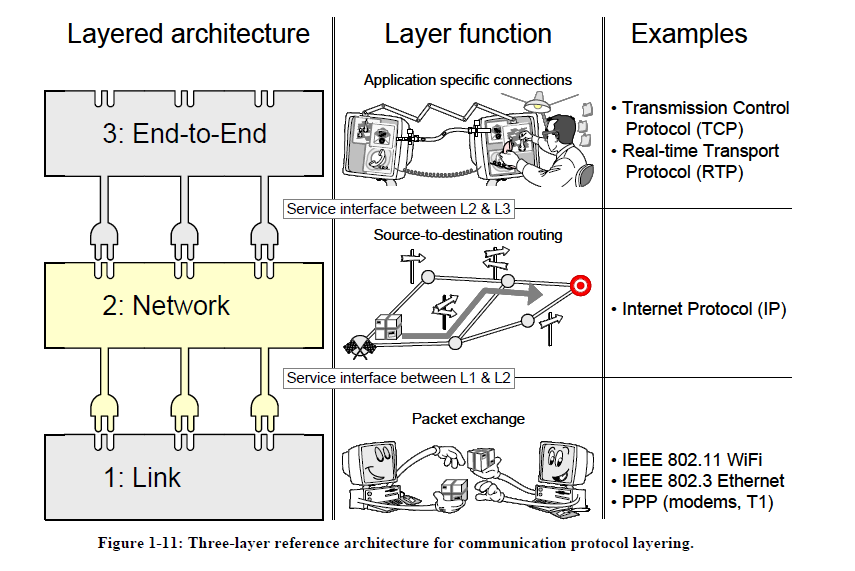
**Describe the three layered architecture**

According to Marsic (2010), the 3 layer architecture for networks has become a standard for developing communication protocols for computer networks. As it's name implies, the architecture has three layers, which are depicted in the following diagram:



(Marsic, 2010).

The first layer at the bottom of the protocol stack is called the LINK layer. This layer is responsible for delivering data packets between nodes via a physical link. The middle layer, called the NETWORK layer, is responsible for connecting one node to another node on a network such that they can exchange data. The third, topmost, layer ia called the END-TO-END layer. It is at this layer that the specific communication application is given consideration. In order for an application to communicate with an application on a different node, packets of data will be encapsulated and sent down through the layer stack, transmitted via the physical medium and then travel back up the stack to the application at the receiving node (Marsic, 2010).

**Briefly explain the difference between link state and distance vector routing**

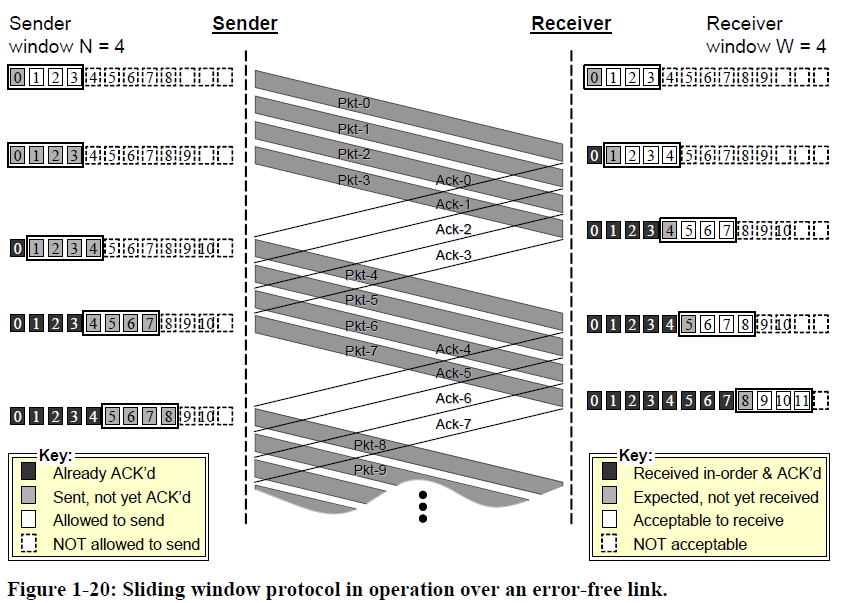
There are several different routing protocols which bring packets to their appropriate destinations. Rather than taking circuitous routes, it would be helpful to send data on the most direct or quickest pathway. Link state routing and distance vector routing are two routing protocols which attempt to find the most efficient pathway (Marsic, 2010).

In link state routing, each node in the network knows the topology of all the nodes in the network (Marsic, 2010). Each node will broadcast the cost of pathways that it is networked to locally. Additionally, each node will create a routing table based on information from other nodes. In this way, not only do the nodes have visibility of all the other nodes on the network, but they have knowledge of the path cost of getting data to each of the other nodes on the network. In this way, the shortest path from one node to another can be computed. Although this method converges quickly on the correct pathway, it requires a lot of resources to maintain and continuously update the routing tables. Moreover, the broadcast nature of disseminating information about the topology between nodes create a lot of network traffic.

In distance vector routing, the entire network topology is not known by each node. Instead, each node only communicates with the node it is in direct contact with (Marsic, 2010). When deciding which node to send a packet through, a node will make the decision based on the shortest path toward the destination node. Furthermore, each node makes the assumption that its neighboring node will use the path that leads to the shortest overall path to the destination. While this sounds simple enough, distance vector routing can lead to routing loops, where two nodes each think that the other one is the best “next” node. Any packet that hits either of these two nodes will subsequently bounce between the two nodes indefinitely.

**Describe how the sliding window protocol works**

Transmission over networks is not infallible. When errors or losses in data transmission occur and are detected, repeating the transmission is often a solution (Marsic, 2010). In the sliding window protocol, a sender will buffer packets and transmit them over the network until the maximum number of unacknowledged packets allowed has been reached. In other words, a sender will keep in buffer any un-acknowleged packets until they have been acknowledged just in case they need to be retransmitted. However, that buffer can only be a certain size and that buffer size will determine how many packets can be “in flight” (or sent but not acknowledged). This protocol is efficient because the sender can continue transmitting (some) data while it waits for acknowledgements of previously sent packets Marsic, 2010). Here is a diagram showing how there is overlap between the ACK responses and subsequent packets in a sliding window protocol with a window size of 4:



(Marcis, 2010).

**Explain the CIDR concept and its major benefit**

Classless Interdomain Routing, or CIDR, became a necessary addressing scheme as the internet grew (Marsic, 2010). Originally, IPv4 addresses had been divided up in to 3 network classes: A, B, and C. Class A networks had the largest number of addresses, and class C had the smallest number of addresses. Since most organizations needed more than the 128 addresses that a class C network has. As such, most organizations were assigned class B networks; but, this was wasteful. Class B networks can have 65,536 addresses, most of which would go unused. Since class B was assigned the most often, not only were they at risk of running out, but the excess addresses in those networks were being wasted.

CIDR reduces wasted addresses by assigning organizations contiguous subsets of the Class C addresses. Instead of assigning an entire class B network to an organization. The organization is assigned a subnet of the Class C network. This is achieved by assigning an address space and mask (A/m). For example, the network 192.206.0.0/21 has addresses ranging from 192.206.0.0 to 192.206.7.255 (Marsic, 2010). The size of the mask can be adjusted to fit the addressing needs of the organization and therefore reduce waste.

References

Marsic, I. (2010). Computer Networks: Performance and Quality of Service. Retrieved from <http://eceweb1.rutgers.edu/~marsic/books/CN/book-CN_marsic.pdf>